

부분적 시각화와 수정을 위한 TIN 압축*

박동규[○] 조환규

부산대학교 전자계산학과 그래픽스 응용 연구실

TIN Compression for Partial Visualization and Modification

Dong-Gyu Park[○] Hwan-Gue Cho

Graphics Application Lab., Dept. of Computer Science, Pusan National University

{dongupak, hgcho}@pearl.cs.pusan.ac.kr

요 약

대부분의 다면체 모델 데이터의 표현과 지리 정보 시스템을 위한 지형 데이터는 삼각 분할을 통하여 만들어진 불규칙 삼각형 네트워크(Triangulated Irregular Network: TIN) 구조를 가지고 있다. 대용량 TIN 데이터를 그래픽스 시스템에서 빠르게 시각화 하는데는 많은 어려움이 있으므로, 일반적으로 TIN 데이터의 특징을 이용하여 압축된 데이터를 이용한다. 그러나 압축된 TIN 데이터는 부분적인 수정과 같은 연산을 수행하기 위하여 전체 데이터를 디코딩하고 다시 인코딩 하는 과정을 반복하여 수행하므로 이같은 연산이 자주 발생하는 응용 분야에는 부적합하다.

본 논문은 이러한 문제의 해결 방안으로 삼각형 메쉬를 삼각형 스트립으로 재구성 한 후에 이 삼각형 스트립을 정점 체인과 각 정점의 차수 정보를 이용하여 저장하는 압축 알고리즘과 함께 이 알고리즘이 부분적인 불규칙 삼각망 수정에 유용하게 적용될 수 있음을 제시한다. 제안된 알고리즘은 각각의 정점 체인이 독립적으로 인코딩 가능하므로 불규칙 삼각망 정보를 수정할 경우 최소한의 인코딩과 디코딩으로 불규칙 삼각망을 수정할 수 있다.

1. 소개

가상 현실과 같은 그래픽스 시스템과 지리 정보 시스템과 같은 응용분야에서 사용되는 TIN 데이터는 일반적으로 수십기가 바이트(giga byte)에서 수 테라 바이트(tera byte)에 이르는 매우 방대한 용량을 차지한다. 이러한 대용량 TIN을 효과적으로 저장하고 표현하는 것은 중요한 문제이다.

이러한 대용량 TIN 데이터를 저장하는 디스크 공간을 절약하기 위한 방법으로 TIN의 연결성 정보와 정점 좌표의 정보를 최대한 적은 양의 데이터로 표현하기 위한 압축 방안이 연구되고 있다[3][5]. TIN의 연결성 압축은 Deering[3], Chow[2], Taubin[8], Floriani[7], Touma[9], Gumhold[4], 김성수[6], 박동규[1] 등에 의하여 연구가 이루어져 왔으며, TIN의 삼각형 연결의 특징을 최대한 이용하여 그 정보를 저장하는 방식이 연구되었다. 표 1은 여러가지 압축 알고리즘의 성능을 분석한 표이다.

실제 지리 정보 시스템과 같은 응용 분야에서는 불규칙 삼각망 정보를 자주 수정하는 일도 발생하는데, 현재 제안된 대부분의 압축 알고리즘으로 압축된 TIN은 압축된 상태로 삼각망의 정보를 수정할 수가 없기 때문에 전체를 디코딩한 후 정보를 수정하고 다시 인코딩한다. 연결성 정보의 압축률이 비트당 0.3 이하로 가장 높은 알고리즘인 김성수의 방식은 임의의 불규칙 삼각망 연결성 정보가 델로니 삼각 분할과 유사하다는 특징을 이용하여 압축하므로 압축과 압축 해독시에 최소한 $O(n \log n)$ 시간이 요구되며 중간 과정에서 많은 메모리가 필요하다.

Deering이 제안한 Generalized Triangle Mesh(GTM)와 같은 압축은 압축과 해독을 비교적 빠르게 수행할 수 있지만 메

쉬 버퍼라는 임시 버퍼를 사용하므로 삼각망의 원하는 영역의 일부만을 시각화하는 것이 불가능하다[3]. 따라서 현재의 압축 알고리즘으로 압축된 삼각망 데이터는 부분적인 수정과 같은 연산을 수행하기 위하여 전체 데이터를 디코딩하고 다시 인코딩 하는 과정을 반복하여 수행하므로 부분적인 수정이 자주 발생하는 응용 분야에는 부적합하다.

2. 기존 연구

TIN 데이터를 저장하기 위한 일반적인 방법중의 하나인 GTM은 Deering에 의하여 연구되었다[3]. 이 방식은 현재 Java 3D API에서 지형 정보를 저장하기 위한 방법으로 사용되고 있다. GTM 압축 방법은 삼각형 스트립을 이용하는데, 이때 이미 참조한 정점을 다시 참조하기 위하여 메쉬 버퍼라고 하는 작은 큐(queue)를 가진다. Chow는 Deering이 사용한 메쉬 버퍼의 성능을 몇가지 휴리스틱을 사용하여 더욱 개선하였다[2].

박동규는 TIN 데이터를 정점 체인과 각 정점 차수를 이용하여 압축하는 방법을 제안하였다[1]. 이 방법은 불규칙 삼각망을 삼각형 스트립으로 재구성 한 후, 삼각형 스트립의 연결성 정보를 정점 체인과 각 정점의 차수만을 이용하여 압축하는 방식으로 그 압축 알고리즘은 그림 1에서 설명되어 있다.

이 압축 방법은 그림 1의 삼각망을 최종적으로 표 2와 같이 저장한다.

그림 1과 표 2에서 보는 바와 같이 삼각형 스트립에서 임의의 정점에 대한 차수는 이전 정점 체인과 연결된 에지(edge)의 개수만을 가지고 구하기 때문에, 정점 차수는 지역성(locality)을 가진다. 이상의 특징을 이용하면 정점의 평균차수의 빈도를 허프만 코딩과 같이 엔트로피 인코딩을 하여 압축할 수 있다.

*본 연구는 1999년도 한국 학술진흥재단 대학부설 연구소 과제 연구비에 의하여 연구되었음

	bits/vertex	압축 방식	점진적 전송	정점 압축
Deering[3]	$1/8 \log n + 8$	GTM	지원 안함	delta encoding
Taubin[8]	4	topological surgery	지원 안함	linear prediction
Hoppe[5]	$\log n + 5$	progressive mesh	지원함	-
Chow[2]	$1/8 \log n + 8$	GTM	지원 안함	variable quantization
Touma[9]	1.5	active list	지원 안함	parallelogram prediction
Gumhold[4]	1.7	cut-border	지원 안함	-
Kim[6]	0.23	Delaunay rule	지원함	-

표 1: 불규칙 삼각망 압축 알고리즘의 성능 분석.

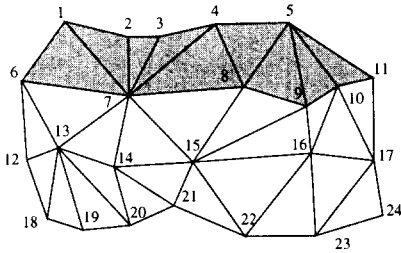


그림 1: 경계 정점 체인 < 1,2,3,4,5 > 로부터 삼각형 스트립을 구성하는 과정

정점 체인	정점 체인 리스트	정점 연결 차수
C^0	< 1, 2, 3, 4, 5 >	< 0, 0, 0, 0, 0 >
C^1	< 6, 7, 8, 9, 10, 11 >	< 1, 4, 2, 1, 1, 1 >
C^2	< 12, 13, 14, 15, 16, 17 >	< 1, 2, 1, 3, 2, 2 >
C^3	< 18, 19, 20, 21, 22, 23, 24 >	< 2, 1, 2, 2, 2, 2, 1 >

표 2: 정점 체인과 각각에 대한 연결 차수로 재구성된 메쉬

실험에 의하면 박동규의 압축 알고리즘은 10만개의 면을 가진 데이터에 대하여 정점당 6.75 비트의 압축률을 보이며 정점의 갯수가 증가함에 따라 압축률 역시 증가하고 있다. 이 결과는 $O(1/8[\log_2(n)] + 8)n$ 의 압축률을 가지는 GTM 방식에 비교하여 높은 압축률을 가짐을 알 수 있다. 본 논문은 정점 체인과 차수를 이용한 TIN 압축 알고리즘이 지역적 수정과 부분적 시각화에 적용될 수 있음을 보인다.

3. 지역적 TIN 수정

지형이나 모델링 데이터의 정보는 새로운 정점의 삽입과 삭제로 인하여 수정된다. 이때 하나의 정점이 삽입되거나 삭제될 경우에 매쉬의 압축된 구조를 재구성한다면 이는 비효율적이다. 현재까지 구현된 TIN 압축 기법은 TIN 내부에 새로운 정점이 삽입되거나 삭제될 경우 전체를 디코딩하여 초기 메쉬 M_0 로 만든 후 그 정보를 갱신한 후 다시 압축하여 저장한다.

그림 2의 메쉬 정보는 GTM 방식에 의하여 인코딩 될 때 "R5p, O1, O6p, O2, O7p, O3, O8p, O4, O9, M13, O-1, O12, O-2, O11, O-3, O10, O-4" 와 같이 저장되어 있다. 이 때 그림 3과 같이 임의의 정점 v_p 가 정점 7과 8사이에 삽입되어 메쉬가 재구성될 경우, 전체 메쉬를 디코딩 하여 각각의 삼각형 중에서 어느 삼각형에 v_p 가 포함되어 있는가 검사를 수행하여야 한다. 검사를 수행하여 정점 v_p 가 삼각형 $\triangle(7, 8, 9)$ 에 있음을 알게되면 삼각형 $\triangle(7, 8, 9)$ 와 $\triangle(12, 7, 9)$ 를 재구성 하여 전체 메쉬를 다시 인코딩 한다. 따라서 전체 메쉬의 인코딩 정보는 "R5p, O1, O6p, O2, O7p, O3, O v_p p, M8p, O4,

O9, M13, M-1, O12, O-2, M-3, O11, O6, O10, O-4" 가 된다.

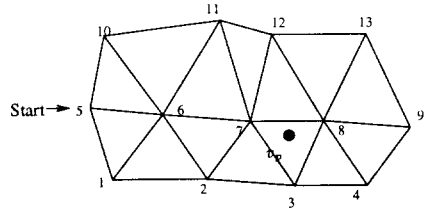


그림 2: 정점 v_p 가 삽입되기 이전 메쉬

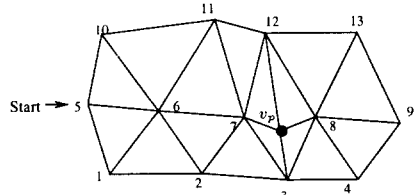


그림 3: 정점 v_p 가 삽입되어 재구성된 메쉬

GTM방식의 압축과 달리 정점 차수를 이용한 압축은 전체 메쉬의 정보를 디코딩 하여 인코딩 하지 않고 전체 메쉬에서 정점이 포함된 정점 체인만을 알면, 최소한의 정보만으로 삼각형 메쉬의 정보를 변경할 수 있다. 즉 임의의 메쉬에 한 정점 v_p 가 삽입될 경우 한 정점이 포함된 정점 체인 C^k 와 C^{k+1} 을 얻은 후 두 정점 체인 중 어느 부분에 새로운 정점 v_p 가 삽입되어야 하는가를 결정한다. 삽입될 위치와 그 연결 차수가 결정되면 정점 체인 C^k 와 C^{k+1} 의 정보를 수정한다.

그림 4와 같이 정점 체인 C^{k-1} 와 C^k 사이에 새로운 정점 v_p 가 삽입되기 전의 정점 체인은 C^k 은 $\langle \dots, v_i^k, v_{i+1}^k, v_{i+2}^k, v_{i+3}^k, \dots \rangle$ 정점 으로 이루어져 있으며 그 각각의 정점 차수는 $\langle \dots, 3, 1, 3, 2, \dots \rangle$ 이다. C^{k+1} 은 $\langle \dots, v_j^{k+1}, v_{j+1}^{k+1}, v_{j+2}^{k+1}, v_{j+3}^{k+1}, v_{j+4}^{k+1}, \dots \rangle$ 정점 으로 이루어져 있으며 그 각각의 정점 차수는 $\langle \dots, 2, 1, 2, 2, 2, \dots \rangle$ 이다.

그림 5와 같이 새로운 정점 v_p 가 삽입되어 정점 체인 C^k 가 재구성 되면 C^k 의 정점 리스트에 v_p 가 추가되고, 이로 인하여 v_p 의 정점 차수가 정점 차수 리스트에 삽입된다. 따라서 정점 체인 C^k 은 $\langle \dots, v_i^k, v_{i+1}^k, v_p, v_{i+2}^k, v_{i+3}^k, \dots \rangle$ 가 되며 그 리스트는 $\langle \dots, 3, 1, 1, 3, 2, \dots \rangle$ 가 된다.

C^k 에 새로운 정점이 추가되면 C^{k+1} 의 정점 차수도 변한다. 따라서 C^{k+1} 의 정점 v_{j+2}^{k+1} 의 정점 차수는 3이 되어서 그 정점 차수는 $\langle \dots, 2, 1, 3, 2, 2, \dots \rangle$ 가 된다.

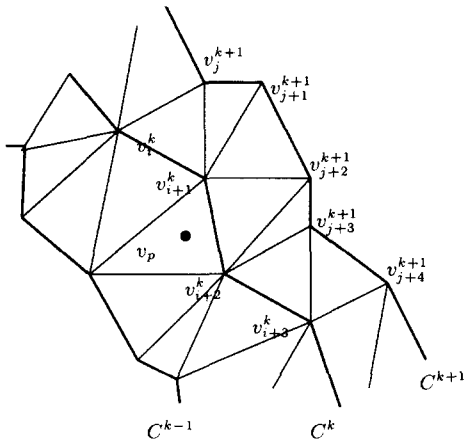


그림 4: 정점 체인 C^{k-1}, C^k, C^{k+1} 에 정점 v_p 가 삽입된 메쉬

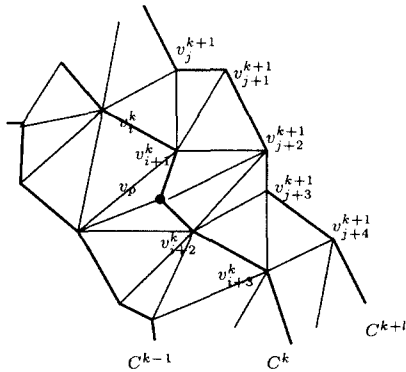


그림 5: 정점 체인 C^k, C^{k+1} 이 재구성된 형태

정점의 삭제는 정점 삽입 과정의 역과정으로 진행되며, 이와 같이 정점 체인과 차수를 이용하면 최소한의 데이터 갱신과 디코딩만으로 정점의 삽입과 삭제를 할 수 있다.

4. 부분적 시각화

정점 차수를 이용한 압축은 지형 데이터의 부분적 시각화가 가능한 특징이 있다. 만일 주어진 지형에서 정점 v_p 와 그 이웃한 삼각형 메쉬에 대한 부분적인 시각화를 하고자 할 경우 기존의 알고리즘은 전체 삼각형 메쉬를 복구한 상태에서 정점에 관한 위치 연산을 수행하여 메쉬를 시각화 한다. 그러나 시각화 하고자 하는 영역이 전체의 일부분에 불과한 경우가 많을 경우 많은 시간과 메모리가 낭비된다.

정점 체인의 부분적인 디코딩이라는 특징을 이용하면 전체 메쉬를 디코딩하지 않고도 최소한의 디코딩만으로 삼각형 메쉬의 시각화가 가능하다.

그림 6와 같은 공간에서 정점 v_p 와 시각화 영역 A_v 을 시각화 하기 위해서는 C^k 과 C^l 내의 정점 체인만을 디코딩하면 된다. 따라서 C^k 이전 정점 체인과 C^l 이후의 정점 체인은 디코딩 하지 않으므로 디코딩 시간과 메모리를 절약할 수 있다.

현재 시점에서 가시화된 영역이 A_v 일 경우 비행 시물레이션은 비행체의 진행 방향으로 가시권이 변화하며, 시물레이션 소프트웨어와 같이 대부분의 시각화 프로그램에서 가시화 영역은 인접한 영역으로 단계적으로 변화한다. 이 변화는

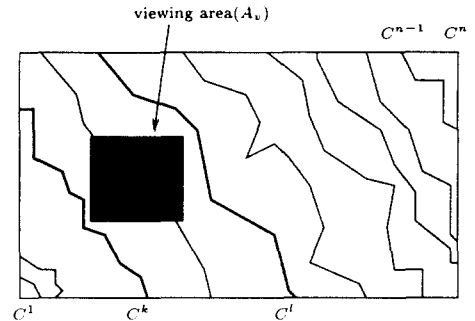


그림 6: 2.5차원 메쉬에서 지정된 영역의 시각화

점진적으로 이루어 지는데 현재 가시화된 정점 체인의 이웃 영역으로 변화한다. 이러한 지역성(locality)로 인하여 제한적인 디코딩만으로 빠르고 효과적으로 시각화가 가능하다.

5. 결론

본 논문은 정점 체인과 그 차수를 이용하여 압축된 불규칙 삼각망을 부분적으로 수정하거나 시각화 하기 위한 방법을 제시한다. 정점 체인과 그 차수를 이용한 메쉬 압축은 Deering이 제안한 GTM에 비하여 압축률이 높으며, 디코딩이 간단한 하드웨어 논리회로로 쉽게 구현가능하며, 각각의 정점 체인이 독립적으로 존재한다.

이러한 특징으로 인하여 정점 체인과 그 차수를 이용한 TIN 압축은 병렬 시각화와 부분적인 가시화 그리고 부분적인 메쉬 수정이 가능한 특징이 있다. 따라서 불규칙 삼각망의 정보가 자주 갱신되는 경우 최소한의 압축과 해독만으로 메쉬의 정보가 수정되는 좋은 특징이 있다. 본 논문은 압축된 불규칙 삼각망에 대한 정보의 갱신이 빈번히 발생할 경우 이를 해결하기 위한 효율적인 메쉬 수정 방안과 시각화 방안을 제시하였다.

참고문헌

- [1] 박동규, 김양수, 조환규. 정점 차수와 삼각형 스트림을 이용한 삼각형 메쉬 압축 기법. 한국 정보과학회 봄 학술발표논문집., pages 650-652, April 1999.
- [2] M. Chow. Optimized geometry compression for realtime rendering. In *IEEE Visualization '97 Proc.*, pages 347-354, Oct. 1997.
- [3] M. Deering. Geometry compression. In *SIGGRAPH '95 Proc.*, pages 13-20, Aug. 1995.
- [4] S. Gumhold and W. Straßer. Real time compression of triangle mesh connectivity. In *SIGGRAPH '98 Proc.*, pages 123-132, July 1998.
- [5] H. Hoppe. Progressive meshes. In *SIGGRAPH '96 Proc.*, pages 99-108, Aug. 1996.
- [6] S. Kim, Y. Kim, M. Cho, and H. Cho. A geometric compression algorithm for massive terrain data using delaunay triangulation. In *WSCG '99*, volume 1, pages 124-131, Feb. 1999.
- [7] P. M. Leila De Floriani and E. Puppo. Compressing tins. In *ACM-GIS'98*, pages 130-145, Nov. 1998.
- [8] G. Taubin and J. Rossignac. Geometric compression through topological surgery. Technical report, Research Report RC-20340, IBM Research Division, 1996.
- [9] C. Touma and C. Gotsman. Triangle mesh compression. In *Proc. of Graphics Interface '98*, pages 26-34, 1998.